

0.1 Technical memo — Layered architecture

Type

Architectural

Factor

How to separate different areas/subsystems so that the coupling among the different subsystems remains as low as possible.

Solution

We use a layered architecture [CITATION: POSA] following the principle of loose layers [CITATION: LARMAN, CHAP 30]. We separate the application into 5 distinct layers: Presentation, Application, Domain, Technical services and Foundation, each with its own area of responsibilities. Each layer will again contain subsystems.

Motivation

This is a classic way of separating high level responsibilities from lower level services, and yields an very robust architecture. It also helps us to have a very distinct separation between the application core and the presentation and application layers. This way we can run the application with different GUIs or with no GUI at all. The latter allows us to write test scripts (feature tests), which are very hard to write through the GUI.

Unresolved issues

Alternatives considered

An architecture based only on model-view-controller [CITATION: POSA] .

See also

Technical memo — model_view_separation.